4. ARCHITECTURE APPLICATION GUIDELINES

This section describes:

The variety of information needed to support the technical and managerial roles of personnel involved in GCCS and the flexibility provided by the various types of architecture depictions

The traceability of requirements among the different architectures

The official coordination and acceptance procedures for GCCS architectural drawings.

DETERMINING YOUR PERSPECTIVE ON GCCS ARCHITECTURE

One of the topics highlighted on in this section is the application of an informal cost-benefit analysis to various techniques for enhancing or updating architecture information. Having read thus far, you may be conducting a little cost-benefit analysis of your own, to the effect of "What exactly do I need to do with these architectures? What am I going to get out of them?"

This section will help you determine exactly how GCCS architectures can benefit you, and on which aspects of the architecture development and management process you need to concentrate your efforts.

The first priority is to sort through the information and options provided up to this point and identify the five or so key architectures that best support the managerial and technical work you need to do.

GCCS development is a complex process requiring the input and decisions of many personnel playing many distinct roles. Typically, these various personnel have diverse tasks, preconceptions, and motivations. The architecture views an engineer finds most valuable are likely to be different from those favored by an end user or a project officer.

There are as many different ways to "slice" GCCS as there are individual perspectives on the system and what needs to be done to further its evolution, as illustrated in Figure 4-1. If you are an engineer, you may be most comfortable with approaching the GCCS in a top-down, hierarchical way, proceeding from general information to more specific, fine-grain details. This perspective is supported by the traditional, core architecture drawings: the functional architecture; the technical architecture; and the system

design/installation/implementation architecture. If on the other hand you are a security manager, your interest may be more or less limited to the security architecture.

Whatever your personal perspective on GCCS, you are likely to see specific parts of the system as most important and worthy of close attention. The various architecture types presented in Section 2 provide an efficient way to highlight particular aspects of the overall system architecture.

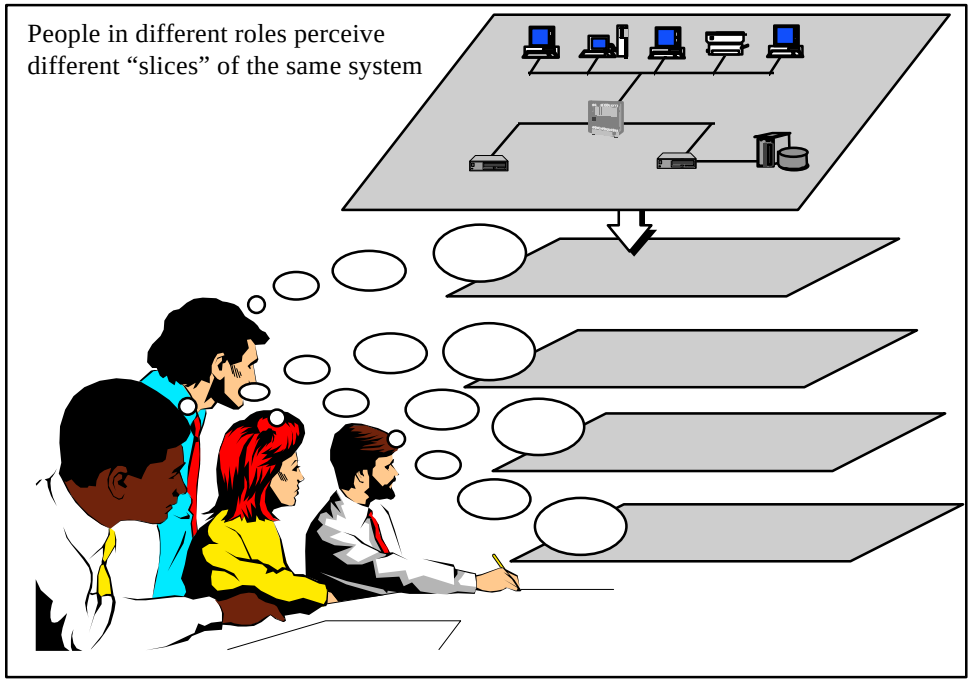


Figure 4-1.
Different "Slices" of the Same System

UNDERSTANDING ARCHITECTURE RELATIONSHIPS

A key to gaining maximum results from architecture work is to understand how your “piece of the puzzle” fits into the whole. With any system, and especially one as complex as the GCCS, there are many pieces of information needed to make sound decisions. These decisions are crucial to the success of the system, and whether you are making those decisions or providing information to support the decision makers, you need to understand the many factors that influence decisions and facilitate or complicate their execution, as shown in Figure 4-2.

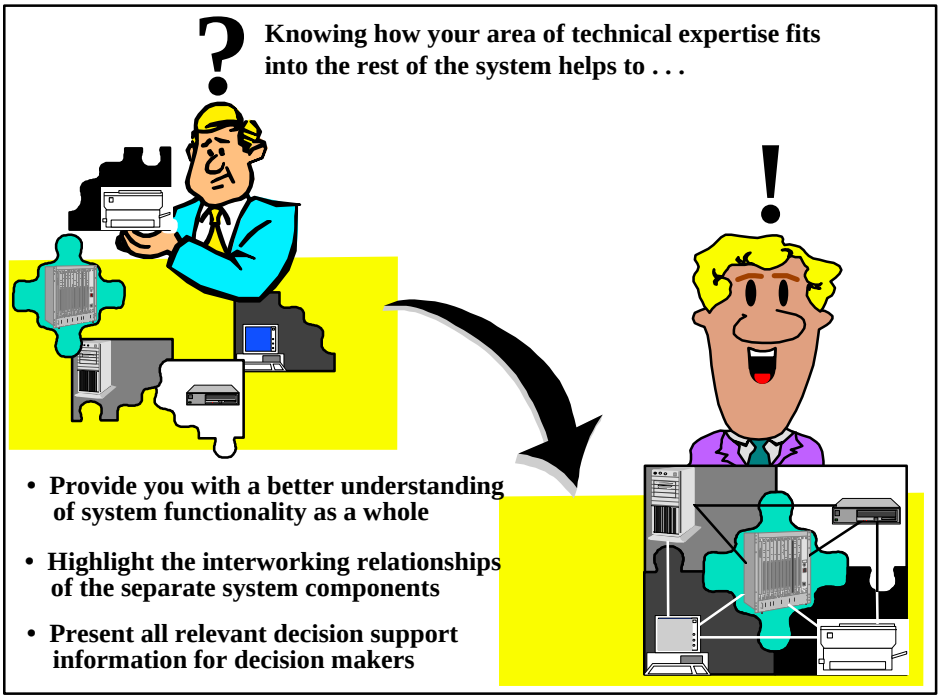


Figure 4-2.
How the Pieces Fit into the Big Picture

Relationships Inherent in Architecture Types

Certain information relationships are inherent in the definitions of architecture types. Taken together, core architectures provide more specific information about the system as a whole than any single architecture type.

The functional architecture depicts the high-level functional and operational requirements and the processes that must occur to achieve the mission. The technical architecture portrays the technical details of how a system will support the functions and processes depicted in the functional architecture. The system design/installation/implementation architecture focuses on the detailed system information that is more generally described in the technical architecture. Using this sequence of architectures, you can trace particular system elements – those features in which you have the greatest interest – throughout the architecture development process.

The hierarchical relationship of the core architectures supports the DoD systems definition and design process. The functional architecture provides a basis for making decisions about the technical environment. These technical decisions are depicted in the technical architecture and, in turn, influence the system design level.

Objective architectures provide a framework for depicting the system objectives and goals. Together, the three objective architectures provide direction for changes to the system. The baseline architecture represents an "As-Is" state of the information system, and is essentially a snapshot of the system as it exists at

the time of the drawing. The target architecture depicts the "To-Be" system, showing the desired system configuration. The difference, or delta, between the baseline and the target architectures represent necessary changes in the system to achieve the desired goal of the system and helps reveal areas requiring special attention. These architectures assist you in planning system migration increments. The information depicted in the baseline and target architectures may be at the level of a functional, technical or system design architecture, depending on the level of detail needed.

The strategic architecture is not necessarily specific to any time frame, but conveys general concepts that communicate key broad objectives for the system. In a strategic architecture, the elements portrayed are often not "to scale." That is, the drawing may depict certain elements of the system at one level of detail and other elements at a different level of detail; either more specific or more abstract. Differing levels of detail are used to focus attention on the key concepts that are being communicated and to downplay extraneous details that do not contribute to the point of the drawing.

Specialty architectures are primarily ad-hoc in nature. They are developed as needed to illustrate or depict information about a system that may not be easily understandable from a core or objective architecture drawing. As represented in the scope versus specificity matrix in Figure 4-3, specialty architectures are not uniformly narrow in scope or specific in detail. For instance, a representational architecture is almost always broad in scope, showing general information at the conceptual level, whereas a security architecture may be highly specific to portray exact security measures to be used by a system.

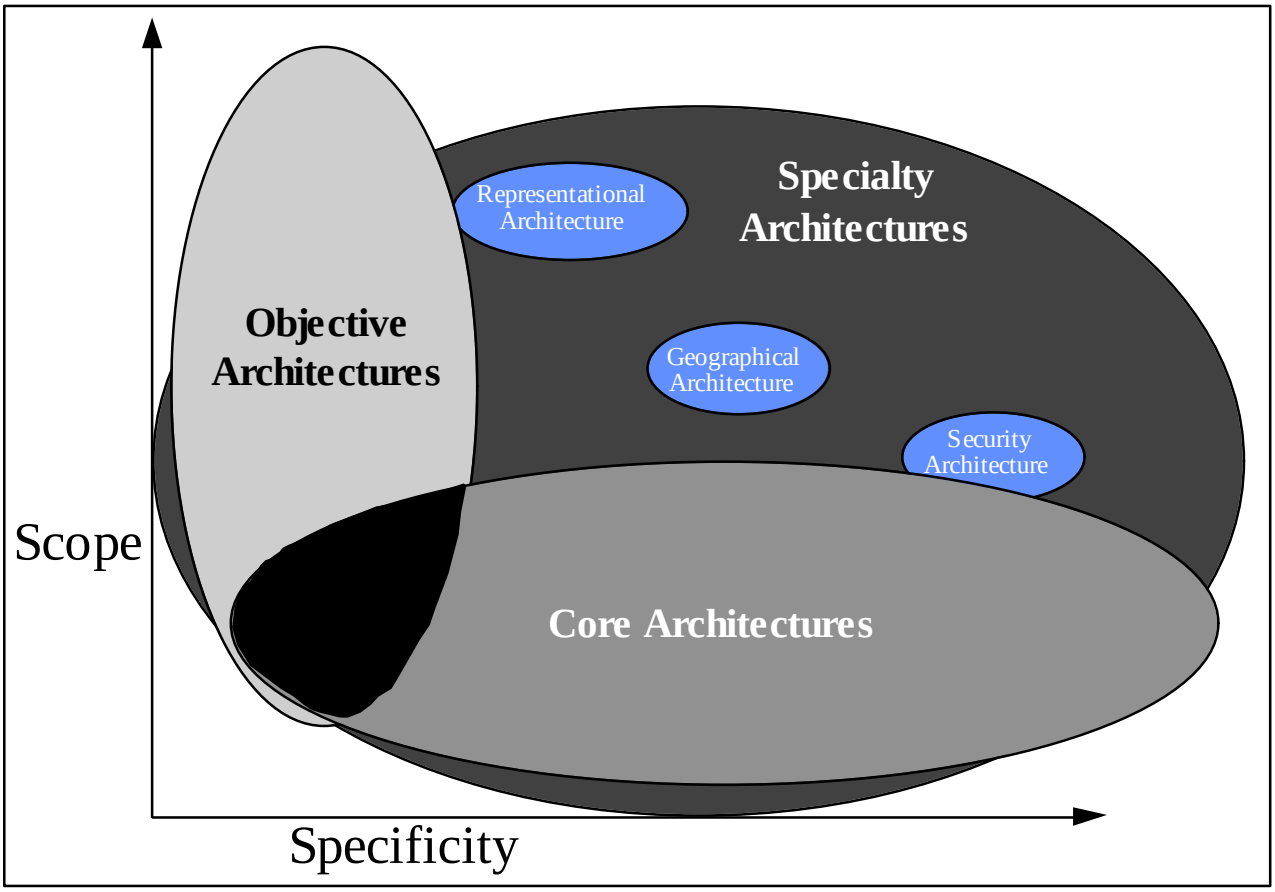


Figure 4-3.
Scope Versus Specificity

AN INFORMATION PRESENTATION OPTION: THEMATIC LAYERING

Once you have determined what types of architectures you need to develop and have considered how those architectures both support and draw from GCCS as a whole, you may want to explore useful approaches to presenting information within an architecture.

Thematic layering is a term borrowed from the mapping discipline. Automated mapping systems typically offer the user the option of turning various layers of the display on and off. Roads, weather, geopolitical boundaries, topography, and other features can be viewed or dropped out of the picture on command.

The General Concept

The application of the thematic layering concept to architecture drawings is illustrated in Figure 4-4. There is an important distinction between different architecture types and thematic layers within an architecture. Each distinct

architecture type (e.g., a baseline or template architecture) is intended to serve as a coherent whole that responds to some managerial or technical need. A thematic layer, on the other hand, simply isolates elements of information to declutter a visual presentation of information. However, the thematic layer is not very useful over the long term by itself.

Some architecture drawings can be segmented into fairly obvious thematic layers. For example, a system architecture drawing could be constructed using layers to depict the network servers and other networking hardware, the network clients, the database servers, the network cabling, and perhaps the security architecture elements. Other architectures may not be so easy to segment.

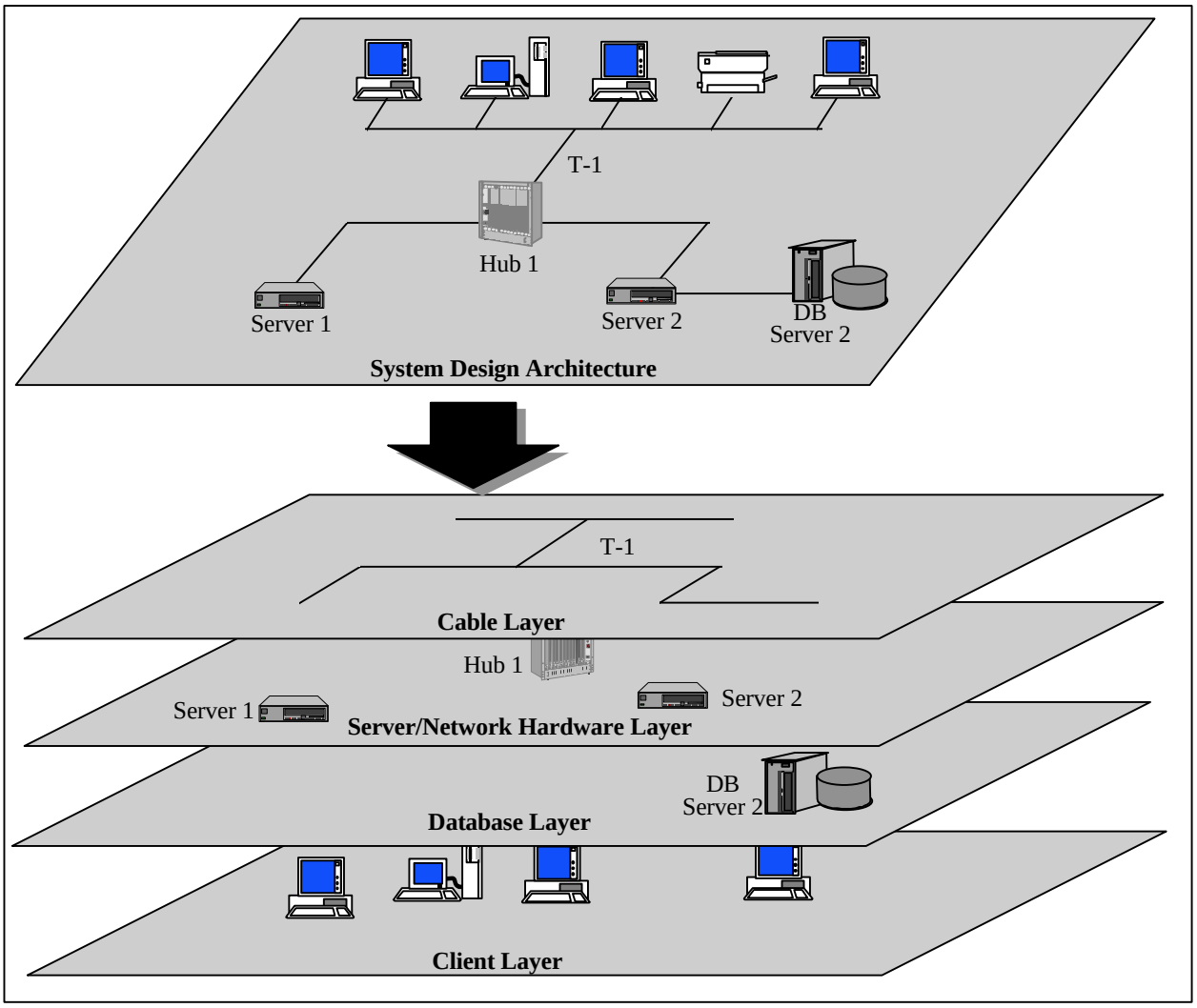


Figure 4-4.
Thematic Layering

As you develop architecture drawings, consider whether thematic layering can help you, and what logical layering schemes might be. Here are some suggestions:

Functional Architecture:

- Functional processes
- Organizations
- Information flow (inputs and outputs)
- Supporting systems.

Technical Architecture:

- Hardware components
- Software components
- Data
- Database management
- Networks
- Security.

System Design/Installation/Implementation Architecture:

- Client computers
- Servers
- Cabling
- Networking devices
- Telecommunications
- Floor plan.

This list is not intended to be comprehensive, but only to give you an idea of the types of information that you may find valuable to separate into thematic layers.

Implementing Thematic Layering

In theory, you could implement thematic layering with minimal support from automated tools. Each layer could be printed on a separate acetate overlay, which could highlight each layer in turn or provide a view of the total system when all the layers were lined up precisely. For most uses, the effort required to prepare the graphics is probably not justified given the appearance and usability of the finished product. Manual production is likely to be disappointing if text or icons overlap at all.

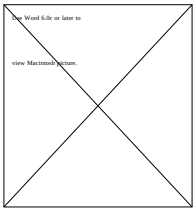
New computer drawing packages are beginning to emerge that provide this layering capability. With such software, the user does not have to worry about keeping the layers in order and exactly lined up. The only complication is assigning drawing elements to the appropriate layers.

Specialized network diagramming tools can make the job even easier. These tools automatically assign cables to one layer, servers to another, and workstations to yet another. They may also offer the capability to customize layers based on security classifications, management factors (e.g., watchstanding shifts), or technical factors (e.g., phased system implementation).

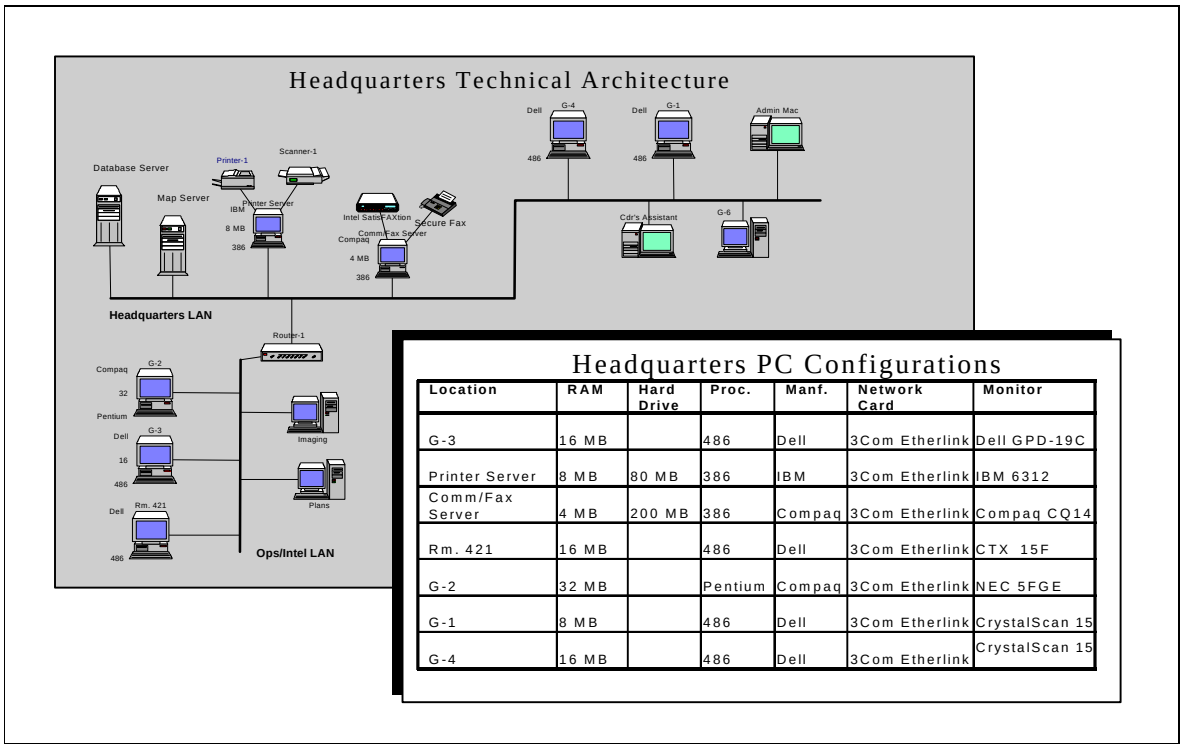
COUPLING BETWEEN ARCHITECTURE DRAWINGS AND SUPPORTING DATA

Quite often, a graphical representation of an information system will not portray all the information necessary to support your requirements. For this reason, it is very useful to have supporting data coupled with architecture drawings. Figure 4-5 shows an example of a system architecture and data that is coupled with it.

The coupling of drawings and supporting data does not necessarily require automated tools, but your job can be made easier by using software products that support this feature. Leading-edge tools expand on the basic listing of attributes next to icons to link graphic elements with fully functional databases.



Capabilities of architecture tools that support thematic layering and database linking can be found in Appendix C. It is also advisable to monitor software vendors' plans to offer enhanced support for these increasingly popular features.



Headquarters PC Configurations

Location	RAM	Hard Drive	Proc.	Manf.	Network Card	Monitor
G-3	16 MB		486	Dell	3Com Etherlink	Dell GPD-19C
Printer Server	8 MB	80 MB	386	IBM	3Com Etherlink	IBM 6312
Comm/Fax Server	4 MB	200 MB	386	Compaq	3Com Etherlink	Compaq CQ14
Rm. 421	16 MB		486	Dell	3Com Etherlink	CTX 15F
G-2	32 MB		Pentium	Compaq	3Com Etherlink	NEC 5FGE
G-1	8 MB		486	Dell	3Com Etherlink	CrystalScan 15
G-4	16 MB		486	Dell	3Com Etherlink	CrystalScan 15

Figure 4-5.
Coupling of Architecture and Supporting Data

DRAWING COORDINATION, ARCHIVAL AND UPDATING

Because of the global nature of GCCS and the fact that there are so many user sites, developers, engineers, architects, integrators, and managers involved in its evolution, coordination of architecture drawings is absolutely essential.

The GCCS Architect has established a hardcopy and electronic archive for GCCS architecture drawings. This archive contains the GCCS architectures for selected sites, as well as source documentation and survey information received from each site andfrom the Services Proponents.

Guidelines for Submittal

All drawings of record and of acceptance should be forwarded to the GCCS architect for review and archival. You may also send any drawing of any type for which a registration number is desired. In addition to the obvious system-wide benefits of gathering all GCCS architecture information into a central repository, the advantages to you of submitting architecture drawings include:

- Assurance that your official documentation is in a safe place. You and your staff may move on to new jobs or your organization may move to new office space. If your local records become lost or jumbled, you can request a copy of your drawings of record from the GCCS Architect.
- Authoritative review of plans and system designs by the GCCS Architect and the GCCS Engineer.
- Official record of agreements. Joint plans, design agreements, or commitments to share resources can be formally documented and maintained by a third party.
- Visibility. Your plans can be used as a reference by other individuals or groups within DoD. Such visibility can lead fresh technical approaches, proposals to combine upgrade or acquisition efforts and share costs, or the exchange of critical lessons learned. In addition, the GCCS Architect's proverbial "seal of approval" and registry number may encourage support for your plan or project elsewhere.

Responsibility of the GCCS Architect

Architecture drawings or supporting materials submitted to the GCCS Architect undergo the following process:

- Initial review to ensure that referenced supporting data is attached
- Review by the GCCS Architect for compliance with GCCS technical standards
- Assignment of a registration number
- Return of materials to sender for major problem resolution, or...

- Detailed review by the GCCS Engineer and Architect (except on those materials for which the GCCS Architect is asked to serve only as an archival point)
- Inclusion in GCCS architecture archive.

The GCCS archive will provide historical information for the GCCS development and will assist configuration management by providing an official, registered set of architectures and data for all GCCS segments.